



# **Ignite's Infobright<sup>®</sup> DB**

## Architecture Overview

WHITE PAPER

# Infobright DB

## Architecture Overview

---

In recent years there has been explosive growth of data from not only Internet activity, but other machine-generated data sources such as computer, network and security logs, sensor data, call detail records, financial trading data, ATM transactions and more.

Ignite's Infobright analytic database platform is a high performance solution for storing and analyzing large volumes of machine-generated data at lower cost and significantly less administrative effort than other database solutions. This paper describes how the Infobright database is architected and the benefits it delivers to enterprises, ISVs and technology providers.

---

# Table of Contents

- 1. Introduction.....4
- 2. Machine-Generated Data .....4
- 3. Rows vs Columns.....5
- 4. The Infobright DB Advantage .....6
- 5. Performance .....7
- 6. Infobright DB Architecture.....8
- 7. Why Infobright DB?.....9
- 8. Infobright DB Technology Details..... 10
- 9. The Granular Computing Engine..... 11
- 10. An Example of Query Resolution Using the Knowledge Grid..... 11
- 11. Data Loading..... 12
- 12. Deployment Options ..... 12

## 1. Introduction

An increasingly connected world driven by Business intelligence (BI) is no longer the sole province of financial analysts and market researchers. Instead, individual lines-of-business and departments that once settled for periodic summary reports now require instant and customized access to business data. As a result, today's leading enterprises are integrating analytic software and processes throughout virtually every operational organization. These enterprises are also demanding that their technology solution providers integrate ad hoc reporting and analysis features within the applications and services they deliver.

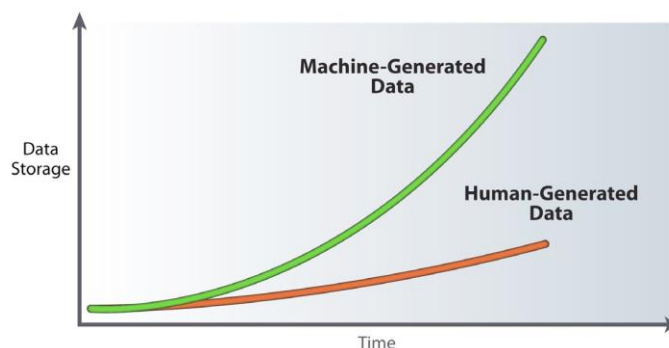
Accompanying the need for on-demand analytics is a significant rise in the growth rate of stored data. The analyst firm Aberdeen points out that storage requirements for BI data now exceed an average annual growth rate of 56 percent year over year and shows no signs of stopping. One data category in particular has emerged as a source of both significant business value and a higher- than-average growth: machine-generated data.

## 2. Machine-Generated Data

Unlike other types of business data, machine-generated data is not constrained by factors such as the number of subscribers or human activity. Instead,

- Produced by computers, sensors and embedded devices
- Typically the result of monitoring functions or observation
- Rarely updated once stored in the database
- Retained for long periods of time for regulatory or governance reasons.

Sources of machine-generated data include Web logs, computer and network events, sensors and RFID-enabled devices, telecom call detail records and automated financial trades. But the most distinguishing aspect of machine-generated data is the volume produced by automated systems. Commenting on this trend, noted database analyst and consultant Curt Monash has written that, "Unlike human-generated data, machine-generated data will grow at Moore's Law kinds of speeds."



This growth reflects the increasing digitization of virtually every aspect of our lives. Yet, it presents enterprises and technology providers with extraordinary technical and business challenges as well as unprecedented opportunities for insight and competitive advantage.

With the growing availability of business data—and machine-generated data in particular— and the advent of Internet of Things applications, leading enterprises are seizing the opportunity to improve decision-making and organizational agility. These needs have given rise to a new operational environment characterized by:

- Users from multiple organizations accessing the same data in different ways
- Dynamic and iterative approaches to data mining, resulting in complex, ad- hoc queries
- The need to have newly-captured data available for analysis in near real- time
- Query results returned in seconds or minutes rather than hours or days

While internal application developers and BI software vendors are already deploying self-service capabilities, many IT organizations remain burdened with growing complexity, costs and time constraints associated with the underlying data management system.

### 3. Rows vs Columns

Many IT organizations and technology solution providers rely on traditional relational databases for their data warehouse, data mart or analytic repository. The problem is that those databases were designed for transactional applications, not analytics against large data volumes. As a result, many companies find that as the volume of data grows, those systems cannot meet the performance requirements from users. In addition, traditional database technology requires a high degree of effort (such as creating and maintaining indexes, creating cubes or projections, or partitioning data) and is costly to license and maintain.

These issues have resulted in the emergence of many new technologies in recent years to address different analytic use cases.

Infobright DB is a columnar database, designed for high performance analytics. Let's examine the difference between a row-oriented database like Oracle, SQL Server, PostgreSQL, or MySQL versus a columnar database.

Row-oriented databases store all values of a data record as one entity. An alternative approach is to store record data in columns. This figure illustrates the difference between row and columnar orientations using employee data as an example.

Employee_ID	Job	Dept	City
1	Shipping	Operations	Toronto
2	Receiving	Operations	Toronto
3	Accounting	Finance	Boston

Data Stored in Rows			
1	Shipping	Operations	Toronto
2	Receiving	Operations	Toronto
3	Accounting	Finance	Boston

Data Stored in Columns			
1	Shipping	Operations	Toronto
2	Receiving	Operations	Toronto
3	Accounting	Finance	Boston

When data records are stored as rows, an application must read each record in its entirety simply to access a single attribute, such as the Department in which employees work. When stored as columns, the database returns only those values associated with the Department attribute. In analytic applications, this approach significantly reduces I/O and lowers query response time, particularly in situations where record sizes are large and users create complex ad hoc queries.

To overcome the limitations associated with row-based architectures, IT organizations have turned to a variety of solutions such as:

- Optimizing databases for specific queries using indexes, cubes and projections
- Upgrading to faster hardware and MPP (Massively Parallel Processing) configurations
- Using offline ETL (Extract, Transform, Load) processes to reduce and consolidate operational data for future analysis.

However, in an enterprise where real-time analytics and self-service BI are critical, these solutions contribute significantly to several operational limitations:

**Impediments to business agility.** Organizations often must wait for DBAs to create indexes or other tuning structures, thereby delaying access to data. In addition, indexes significantly slow data-loading operations and increase the size of the database, sometimes by a factor of 2x.

**Loss of data and time fidelity.** IT generally performs ETL operations in batch mode during non-business hours. Such transformations delay access to data and often result in mismatches between operational and analytic databases.

**Limited ad hoc capability.** Response times for ad hoc queries increase as the volume of data grows. Unanticipated queries (where DBAs have not tuned the database in advance) can result in unacceptable response times, and may even fail to complete.

**Unnecessary expenditures.** Attempts to improve performance using hardware acceleration and database tuning schemes raise the capital costs of equipment and the operational costs of database administration. Further, the added complexity of managing a large database diverts operational budgets away from more urgent IT projects.

Since they access only the data values required to resolve individual queries, columnar databases are becoming the preferred choice for analytic applications. Compared to row-oriented databases, they offer benefits of higher flexibility, performance and data accessibility.

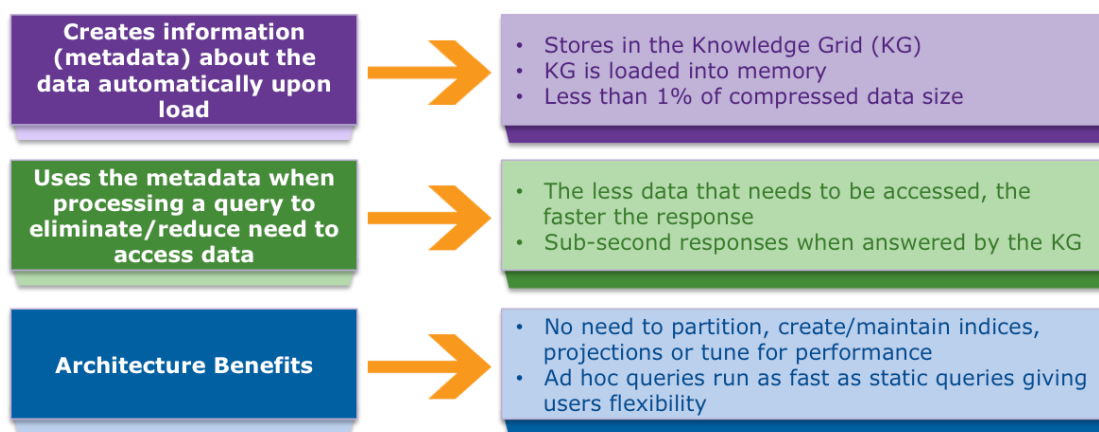
## 4. The Infobright DB Advantage

Although faster than row-based architectures for analytics, many columnar implementations do lose their performance advantage in the face of growing data volumes. This limitation is generally due to the increasing number of I/O operations required to resolve queries as database size increases. Compounding the problem is the high number of data records that are read and discarded because they do not meet the query parameters.

To compensate, some columnar database vendors offer traditional, row-style tuning schemes such as indices or projections. In addition, a number of solutions rely on data partitioning or complex hardware configurations, which require IT organizations to upgrade or add servers as data volumes increase. Such approaches fail to address the operational limitations and need for ad hoc analytics identified above, making them ineffective for near real-time environments with high volumes of machine-generated data.

Infobright DB is specifically designed to achieve high performance for large volumes of machine-generated data used in complex and ad hoc analytic environments, without the database administration other products require. Unlike row-oriented databases or other columnar architectures, Infobright DB combines semantic intelligence with advanced compression technology to speed queries and reduce hardware footprint. Infobright DB is a cost-effective solution designed to ensure on-demand performance without database tuning and administration, while minimizing the amount of required storage and server capacity, even as data volumes grow.

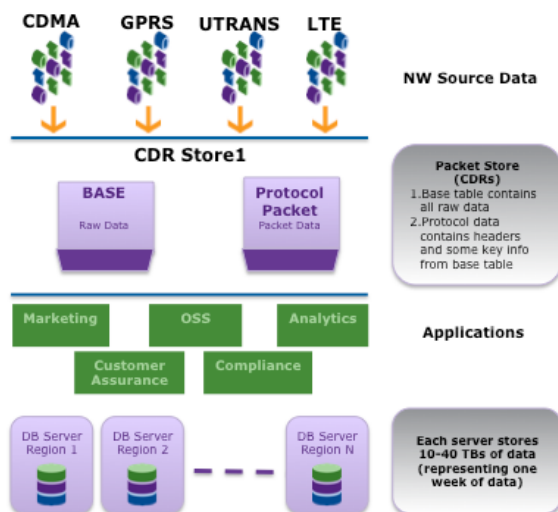
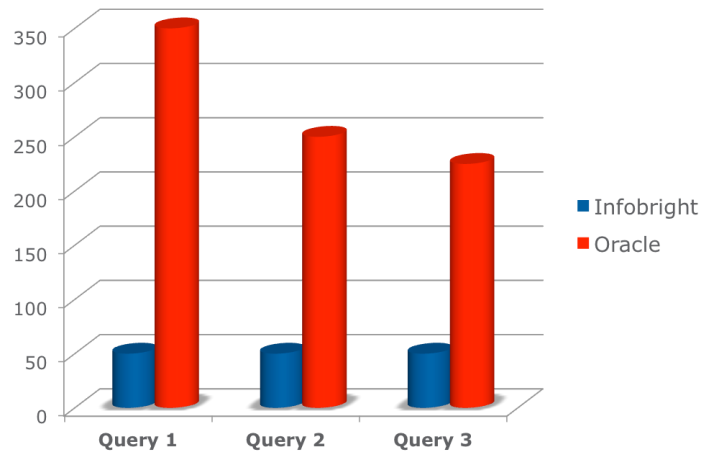
Infobright DB combines a column-orientation with intelligent technology that simplifies administration, eliminates the need to tune for performance, and reduces total costs. Here is a quick overview of how it works:



## 5. Performance

How does Infobright DB perform? Let's look at one example from a mid-sized telecom company. The chart compares the results of their testing using Oracle and Infobright DB. The database contained 771MM rows of call detail records (CDRs). The performance difference is clear. Infobright DB's Knowledge Grid and Granular Engine returned many of the results without accessing the actual CDR data. The Infobright DB solution reduces overall I/O by placing intelligence in the software and not relying on static tuning, hardware acceleration or complex MPP configurations.

In addition to raw execution performance, Infobright DB offers several technical benefits to enterprise end users, IT organizations, DBAs and application developers. Further, its small footprint and low administration requirements make it ideal as an embedded database for ISV and SaaS solutions within data intensive applications such as network management, telecom CDR analysis, or log analytics.



The diagram to the left represents a large scale Infobright DB environment. Load speed for this user exceeded 2TB / hour into a single table and the database size per database instance ranges from 10 – 40TBs of data. The total application will be storing between 700TB – 1.8 PB's on Infobright DB.

## 6. Infobright DB Architecture

Let's examine the main components within the Infobright DB architecture.

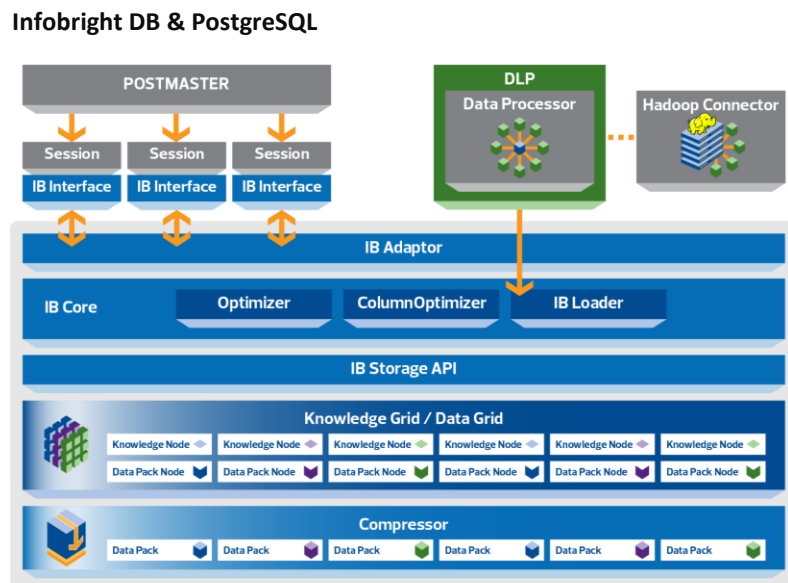
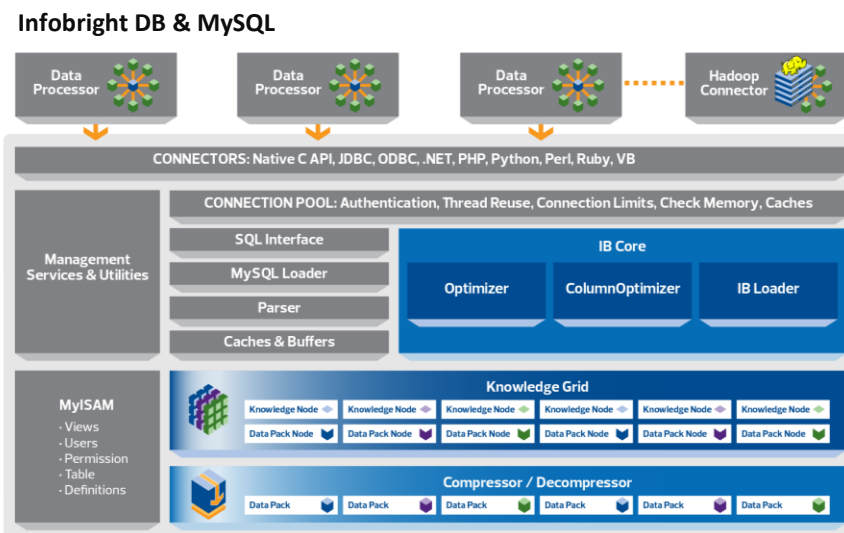
**Data packs and deep data compression** offer a unique approach to data organization. As data is loaded into Infobright DB, it is stored in Data Packs, fixed-sized segments of 65,536 values within each column. Infobright DB compresses each Data Pack individually using the optimal compression algorithm for that data, resulting in very deep data compression, typically 20:1 to 40:1.

**The Knowledge Grid** is an in-memory structure that automatically creates and stores information about the data upon load and when queries are executed. It is used to deliver fast query performance without DBA effort.

**The Granular Computing Engine** processes queries using the Knowledge Grid information to optimize query processing. The goal is to eliminate or significantly reduce the amount of data that needs to be decompressed and accessed to answer a query. Infobright DB can often answer queries referencing only the Knowledge Grid information, (without having to read the data), which results in sub-second query response for those queries.

**High-speed loading options** for Infobright DB that make incoming machine-generated data available to users for near real-time analysis. Infobright DB includes an integrated loader. Also available is an add-on product for Infobright DB, the Distributed Load Processor (DLP), that scales load speed linearly using remote servers. DLP also includes a connector for Hadoop to simplify the process of extracting data from HDFS and loading it into Infobright DB.

### Infobright DB & MySQL



**ColumnOptimizer** extends the Knowledge Grid intelligence by adding information about a particular data domain—such as web, financial services or telecom data—which the Granular Engine uses to optimize further for machine-generated data. Information about online data types such as email addresses, URLs and IP addresses are pre-defined, and users can also easily add their own domain intelligence to meet their unique needs.

**Rough Query** is a unique feature of Infobright DB that can speed up query response time by a factor of 20 or more for investigative analytics against a large volume of data.

Rather than execute a long-running query to find a specific answer (which could even return a null response), Rough Query enables a user to narrow down the results in an iterative manner, with sub-second response time, before the full query is run. In combination, Infobright DB's very high rate of data compression and its Rough Query capability allow companies to store far more data history, yet drill down into the data in a fraction of the time of other databases.



**Approximate Query** takes the notion of Rough Query a step further by evaluating more data; it does not rely on the knowledge grid alone but actually opens a sampled set of data packs. Whereas Rough Query provides an interval to allow for data exploration, Approximate Query provides a response that is in the same form as a normal query. Approximate Query can return results many times faster than are a useful tool for exploring very large data sets.

## 7. Why Infobright DB?

Infobright DB Advantage	How We Do It	Benefits & Proof Points
<b>Query Performance</b>	<p>Store highly compressed data in columns.</p> <p>Create Knowledge Grid information during load and query operations.</p> <p>Eliminate or reduce the need to access data by resolving queries using the Knowledge Grid information.</p>	<p>Sample query results from customer tests:</p> <p>Infobright DB: 46 seconds  Oracle: 5 min 55 sec  Infobright DB: 10 seconds  SQL Server: 11 minutes</p>
<b>Load Performance</b>	<p>Because there are no indexes, load speed remains constant even as the size of the database grows. Multiple loader options provide the features and load speed required for large data volumes.</p>	<p>Load speed varies depending on the loader option used. It may range from:</p> <ul style="list-style-type: none"> <li>50GB per hour using the MySQL loader</li> <li>80-150GB per hour using the Infobright DB loader</li> <li>2TB per hour and more using the Distributed Load Processor (DLP)</li> </ul>
<b>Rapid Installation</b>	<p>Installs in minutes and requires no configuration, no index creation and no schema setup.</p>	<p>Many traditional databases require days or weeks of work configuring the database, creating indices, partitioning data, or creating cubes or projections.</p>
<b>Near Real-time Data Access</b>	<p>Infobright DB ensures new data records are immediately available for complex, ad hoc queries by combining:</p> <ul style="list-style-type: none"> <li>High load speeds with DLP</li> <li>Constant load speed over time</li> <li>No need for DBA effort to tune for faster query response.</li> </ul>	<p>In one of the world's largest telecom networks, call data was loaded at a rate exceeding 1 billions records per hour.</p>
<b>Migration and Compatibility</b>	<p>Support for:</p> <ul style="list-style-type: none"> <li>Solaris, Linux and Windows</li> <li>ANSI SQL and MySQL</li> <li>Broad range of languages and drivers including Java, C/C++, Ruby, Perl, 4th</li> <li>GL's, ODBC and JDBC.</li> </ul>	<p>Works with major BI and ETL tools from vendors such as Actuate/BIRT, Jaspersoft, Pentaho, Talend, MicroStrategy, Cognos, Informatica, Business Objects, and others.</p> <p>Broad support of languages and interfaces using the standard MySQL drivers.</p>
<b>Lower Cost: Fewer Servers</b>	<p>Uses intelligence in the software instead of relying on brute-force CPU power or complex MPP configurations.</p>	<p>Infobright DB has been tested to 150TB of data in a single server environment. Infobright DB supports replication for high availability configurations.</p>
<b>Lower Cost: Less Storage</b>	<p>Compresses data at an average ratio of 20:1 and up to 40:1.</p>	<p>A telecom provider reduced storage requirements from 1.3TB using Oracle to 123GB using Infobright DB.</p>
<b>Lower Cost: Administration</b>	<p>No indexing, data partitioning, data duplication, database tuning or aggregate tables to manage means 90 percent less effort.</p>	<p>Compare this to what you do today.</p>

## 8. Infobright DB Technology Details

Infobright DB's architecture combines a columnar database and a unique Knowledge Grid architecture optimized for analytics. The following is an overview of the architecture and a description of the data loading and compression technology.

### Data Organization and the Knowledge Grid

The Infobright database resolves complex analytic queries without the need for traditional indexes, data partitioning, projections, manual tuning or specific schemas. Instead, the Knowledge Grid architecture automatically creates and stores the information needed to quickly resolve these queries. Infobright DB organizes the data into 2 layers: the compressed data itself that is stored in segments called Data Packs, and information about the data which comprises the components of the Knowledge Grid. For each query, the Infobright DB Granular Computing Engine uses the information in the Knowledge Grid to determine which Data Packs are relevant to the query before decompressing any data.

### Data Packs and Compression

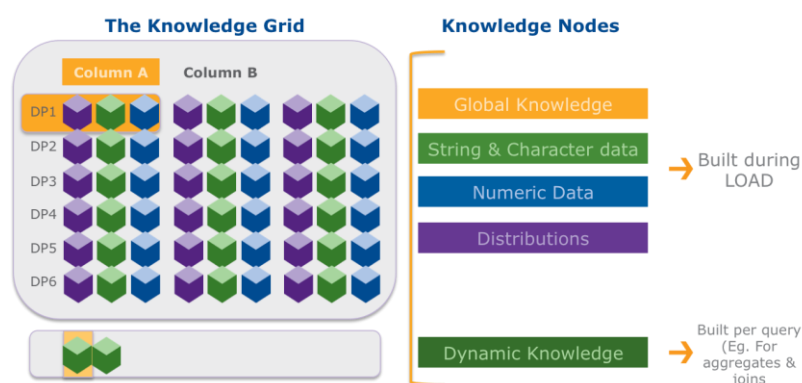
The data within each column is stored in 65,536 item groupings called Data Packs. The use of Data Packs improves data compression as the optimal compression algorithm is applied based on the data contents. An average compression ratio of 10:1 is achieved after loading data into Infobright DB. For example 10TB of raw data can be stored in about 1TB of space, including the 1% overhead associated with creating the Data Pack

### Nodes and the Knowledge Grid

The compression algorithms for each column are selected based on the data type, and are applied iteratively to each Data Pack until maximum compression for that Data Pack has been achieved. Within a column the compression ratio may differ between Data Packs depending on how repetitive the data is. Some customers have reported an overall compression ratio as high as 40:1.

### Data Pack Nodes (DPNs)

Data Pack Nodes contain information about the contents of each data pack. It includes a set of statistics and aggregate values of the data from each Data Pack: MIN, MAX, SUM, AVG, COUNT, and No. of NULLs. There is a 1:1 relationship between Data Packs and the DPNs that are created automatically during load. The Granular Engine then has permanent summary information available about all of the data in the database that will later be used to flag relevant Data Packs when resolving queries. In the case of traditional databases, query resolution is aided by indices that are created for a subset of columns only.



### Knowledge Nodes (KNs)

This is an additional set of metadata that is more introspective of the data within the Data Packs, describing ranges of numeric value occurrences and character positions, as well as column relationships between Data Packs. The introspective KNs are created at load time, and the KNs relating the columns to each other are created dynamically in response to queries involving JOINS in order to optimize performance.

The DPNs and KNs together form the Knowledge Grid. Unlike the indexes required for traditional databases, DPNs and KNs are not manually created, and require no ongoing care and maintenance. Instead, they are created and managed automatically by the system. In essence, the Knowledge Grid provides a high level view of the entire content of the database with a minimal overhead of approximately 1% of the original data. (By contrast, classic indexes may represent as much as 20% to 100% of the size of the original data.)

## 9. The Granular Computing Engine

The Granular Engine is the highest level of intelligence in the architecture. It uses the Knowledge Grid to determine the minimum set of Data Packs needed to be decompressed in order to satisfy a given query in the fastest possible time by identifying the relevant Data Packs. In some cases the summary information already contained in the Knowledge Grid is sufficient to resolve the query, and nothing is decompressed.

How do Data Packs, DPNs and KNs work together to achieve fast query performance?

For each query, the Granular Engine uses the summary information in the DPNs and KNs to group the Data Packs into one of the three following categories:

- Relevant Packs – where each element (the record’s value for the given column) is identified, based on DPNs and KNs, as applicable to the given query,
- Irrelevant Packs – where the Data Pack elements hold no relevant values based on the DPN and KN statistics, or
- Suspect Packs – where some relevant elements exist within a certain range, but the Data Pack needs to be decompressed in order to determine the detailed values specific to the query.

The Relevant and Suspect packs are then used to resolve the query. In some cases, for example if we’re asking for aggregates, only the Suspect packs need to be decompressed because the Relevant packs will have the aggregate value(s) pre-determined. However, if the query is asking for record details, then all Suspect and all Relevant packs will need to be decompressed.

## 10. An Example of Query Resolution Using the Knowledge Grid

Let’s look at a table of employees with the following 4 columns: salary, age, job, and city. Now let’s apply a query that asks for a count of the number of employees fitting a particular description. In this case we want to know the number of employees that have a salary over \$50k, are below the age of 35, have a job description of Accounting and are working out of the Toronto office.

Here is the sample query:

```
SELECT count(*) FROM employees
WHERE salary > 50,000 AND age < 35
AND job = 'Accounting' AND city = 'TORONTO';
```

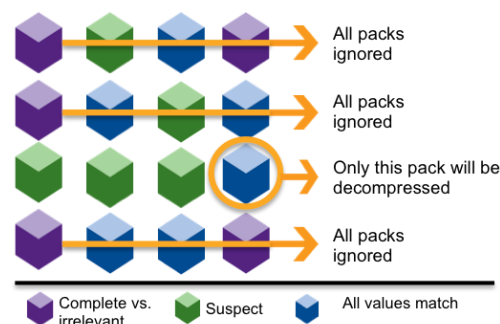
The Granular Engine uses the specific constraints of the query to identify the Relevant, Suspect, and Irrelevant Data Packs.

In our sample table, the first constraint of salary > 50000 eliminates 3 of the 4 Data Packs by using the MIN and MAX information stored in the Data Pack Nodes to tell us that all values in these Data Packs are less than or equal to 50000, making them Irrelevant.

Using similar logic for the column “age”, we can determine that 2 Data Packs contain employees with age under 35, making them Relevant, and 2 Data Packs contain some employees with age under 35 making them Suspect, since we need more details about the data to determine how many. We continue this logic to the job and city columns, but since we’ve already identified Irrelevant Data Packs in the first column based on salary, only the 3rd row of Data Packs for the entire table needs to be examined. We essentially use the Knowledge Grid to eliminate all Data Pack rows that have any columns flagged as Irrelevant.

In this example we found that only the Data Pack of the column “city” actually needs to be decompressed since the other three were found to be Relevant, so of the entire table we now only have one Data Pack to decompress.

That is the key to the fast query response Infobright DB delivers – eliminating the need to decompress and read extraneous data in order to resolve a query.



## 11. Data Loading

Infobright DB includes multiple options for loading data into the database:

- The Infobright DB Loader
- For Infobright DB MySQL Edition: the MySQL Loader
- Distributed Load Processor (DLP) add-on
- Loading data using third party ETL tools

The multi-threaded Infobright DB loader is delivered as part of Infobright DB solution. ICE includes a single-threaded version of the Infobright DB loader.

Since it was designed for fast data loading, the Infobright DB Loader has stricter requirements in terms of data formatting and less error checking than the integrated database loaders (MySQL or PostgreSQL), as it assumes that the incoming data is aligned with the target database table and suitably formatted. The high speed Infobright DB Loader can be used for both text and binary files, achieving up to 80GB per hour for text files, and up to 150GB per hour for binary loads on a single server with parallel loads.

The Distributed Load Processor (DLP) is an add-on product for Infobright DB 4.0 and beyond. DLP scales load speed linearly across multiple servers by remotely processing and compressing data, building the Knowledge Nodes, and then transferring the compressed data to the Infobright database. Performance gains result because CPU-intensive data compression is distributed across machines. Load speeds of over 2TB per hour can be achieved based on the degree of scale-out using multiple servers.

DLP also provides connectivity to a Hadoop cluster, letting users easily leverage Hadoop's large-scale distributed batch processing benefits with Infobright DB's fast ad hoc analytic capabilities. The Hadoop connector provides a simple way to extract data from the HDFS (Hadoop Distributed File System) and load it into Infobright DB at very high speeds.

## 12. Deployment Options

Infobright DB's initial products were MySQL based and available in an open source and a commercial version. As the demand for high performance analytic databases has grown, so have customer requirements for different deployment options. Today, Infobright DB is available as both software and as a data appliance and supports either MySQL or PostgreSQL databases. The following describes these options in more detail.

Infobright DB MySQL Edition is software built on MySQL and leverages the connectors and interoperability of MySQL standards. This integration with MySQL allows Infobright DB to tie in seamlessly with any ETL and BI tool that is compatible with MySQL and to leverage the extensive driver connectivity provided by MySQL connectors (including ODBC, JDBC, C/C++, .NET, Perl, Python, PHP, Ruby, Tcl and others). MySQL also provides cataloging functions such as table definitions, views, users, permissions, etc. which Infobright DB stores in a MyISAM database. Infobright DB MySQL Edition includes MySQL as part of its distribution. Infobright DB MySQL Edition is supported on a range of Linux and Microsoft Windows operating systems.

Infobright DB PostgreSQL Edition is a software delivery model built on PostgreSQL and provides an Open Source alternative to MySQL. It leverages the connectors and interoperability of PostgreSQL standards allows Infobright DB to tie in seamlessly with any ETL and BI tool that is compatible with PostgreSQL and to leverage the extensive driver connectivity provided by PostgreSQL connectors (including ODBC, JDBC, C/C++, .NET, Perl, Python, PHP, Ruby, Tcl and others). Infobright DB PostgreSQL Edition includes PostgreSQL as part of its distribution. Infobright DB PostgreSQL Edition is supported on a range of Linux and Microsoft Windows operating systems.

If you would like to learn more about Ignite's Infobright DB, or to download a trial evaluation, please visit <http://www.ignitetechnologies.com/solutions/information-technology/infobrightdb>



---

Copyright © 2017 Ignite Technologies, Inc.  
All Rights Reserved.  
401 Congress Ave., Suite 2560  
Austin, TX 78701  
800-248-0027  
[info@ignitetechnologies.com](mailto:info@ignitetechnologies.com)